Selecting Technologies for Long-Term Survival

Martin Holmes
Humanities Computing and Media Centre

Endings project
https://onlineacademiccommunity.uvic.ca/endingsproject/

University of Victoria | Humanities Computing and Media Centre

The problem

“Interactive digital assets are far more complex to preserve and manage than single, uniform digital media files. A single interactive work can comprise an entire range of digital objects, including files in different types and formats, applications to coordinate the files, and operating systems to run the applications. If any part of this complex system fails, the entire asset can become unreadable.”

Rieger and Casad 2014: “Interactive Digital Media Art Survey: Key Findings and Observations.”

Assumptions

Assumptions

- No apocalypse

Assumptions

- No apocalypse
- Bits can be stored and retrieved

Assumptions

- No apocalypse
- Bits can be stored and retrieved
- Archives exist to store and deliver them

Assumptions

- No apocalypse
- Bits can be stored and retrieved
- Archives exist to store and deliver them
- Change, innovation and obsolescence proceed at least as fast as in the last few years

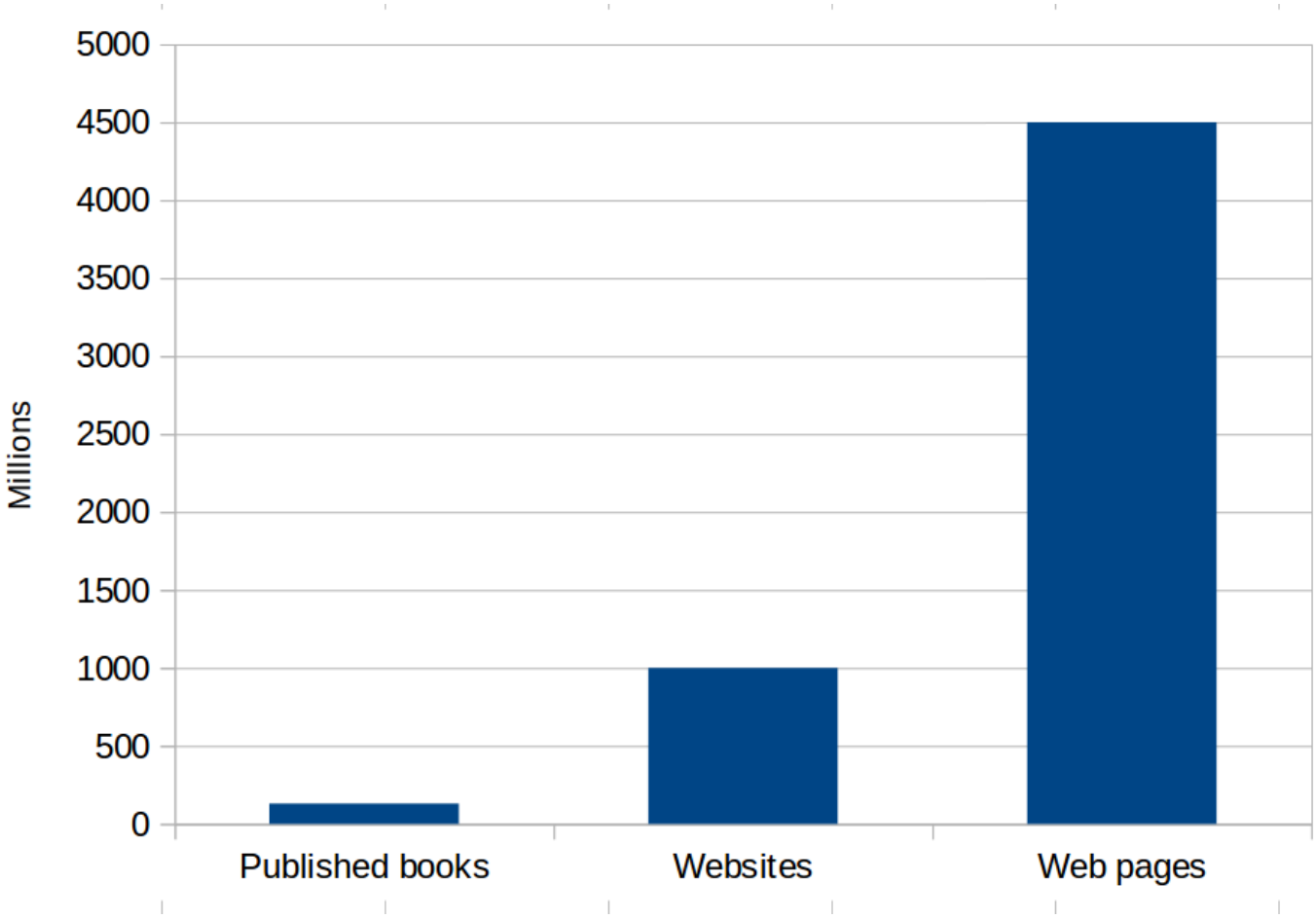
Strategy

- No software requirements
- No network requirements
- Easily-archivable bits
- File formats with *massive* buy-in/investment
- Functionality that gracefully degrades

50 years? Seriously?

Yes, seriously. 50 years.

1. HTML

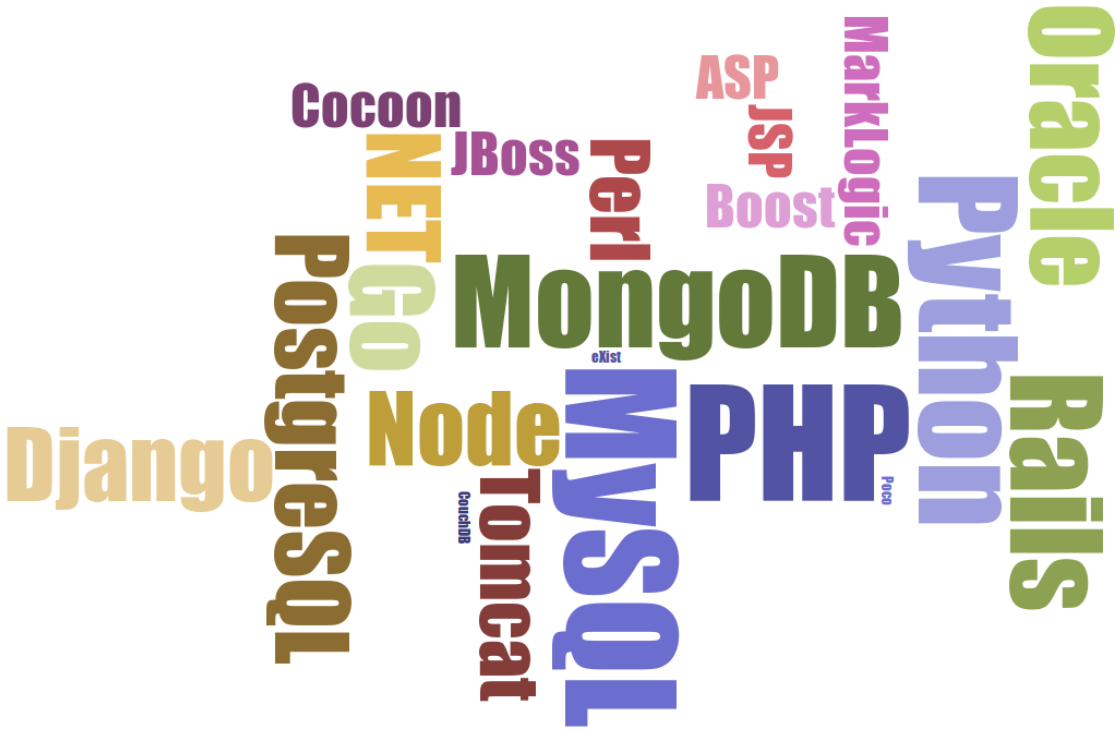


http://www.worldwidewebsize.com/

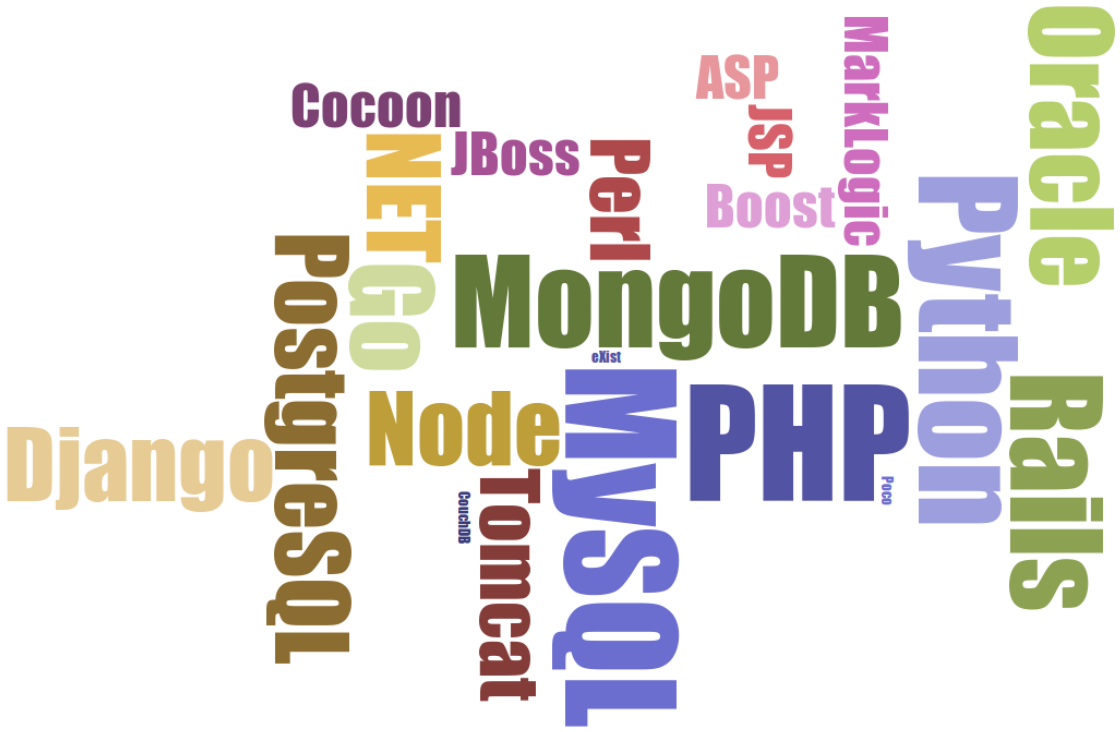
1. HTML

- The world’s first website still works: http://info.cern.ch/hypertext/WWW/TheProject.html
- This is because of *backwards compatibility* and *error tolerance.*

But what about...

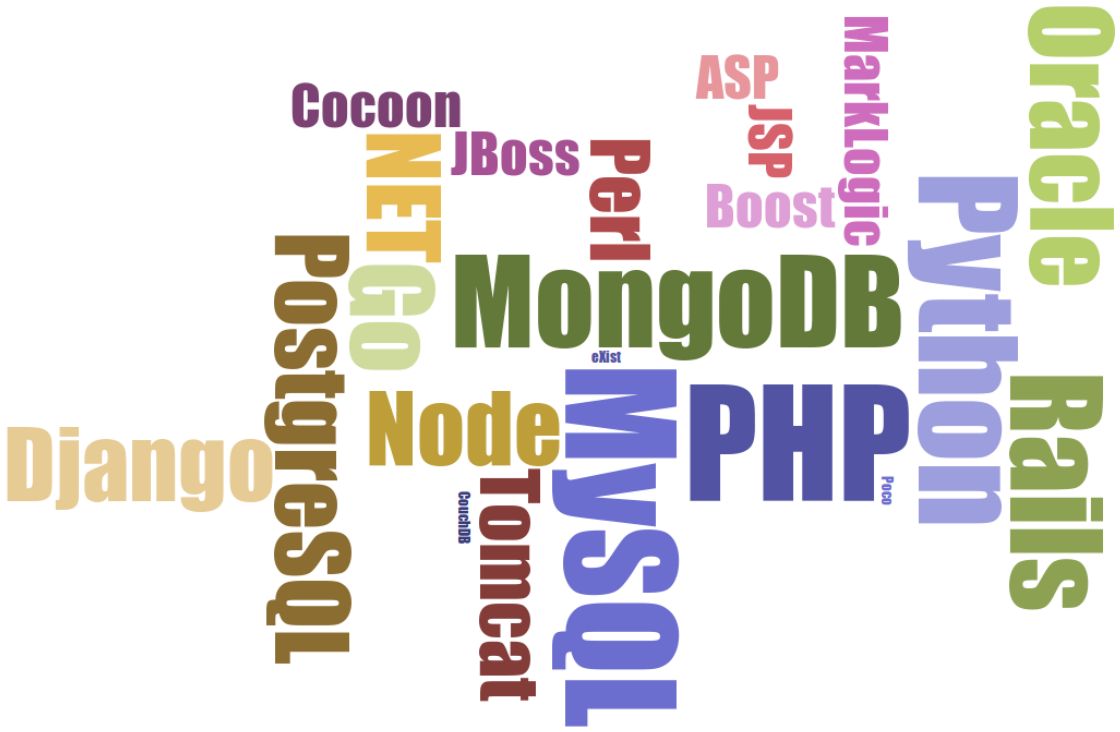


But what about...



What are they mostly for?

But what about...



What are they mostly for?

Producing HTML.

XHTML5

XHTML5

X = processable / transformable with XML tools

XHTML5

X = processable / transformable with XML tools

HTML = 4.5 billion pages

XHTML5

X = processable / transformable with XML tools

HTML = 4.5 billion pages

5 =

XHTML5

X = processable / transformable with XML tools

HTML = 4.5 billion pages

5 =

- 62.5% of web sites

XHTML5

X = processable / transformable with XML tools

HTML = 4.5 billion pages

5 =

- 62.5% of web sites
- robust specification, which

XHTML5

X = processable / transformable with XML tools

HTML = 4.5 billion pages

5 =

- 62.5% of web sites
- robust specification, which
- “defines the parsing rules for HTML documents, whether they are syntactically correct or not”

XHTML5

X = processable / transformable with XML tools

HTML = 4.5 billion pages

5 =

- 62.5% of web sites
- robust specification, which
- “defines the parsing rules for HTML documents, whether they are syntactically correct or not”
- allows custom data

Custom data in XHTML5

```
<p>
    The
    <span data-teiEl="name" data-teiAttType="vessel">
        Titanic
    </span>
    was a mighty ship.
</p>
```

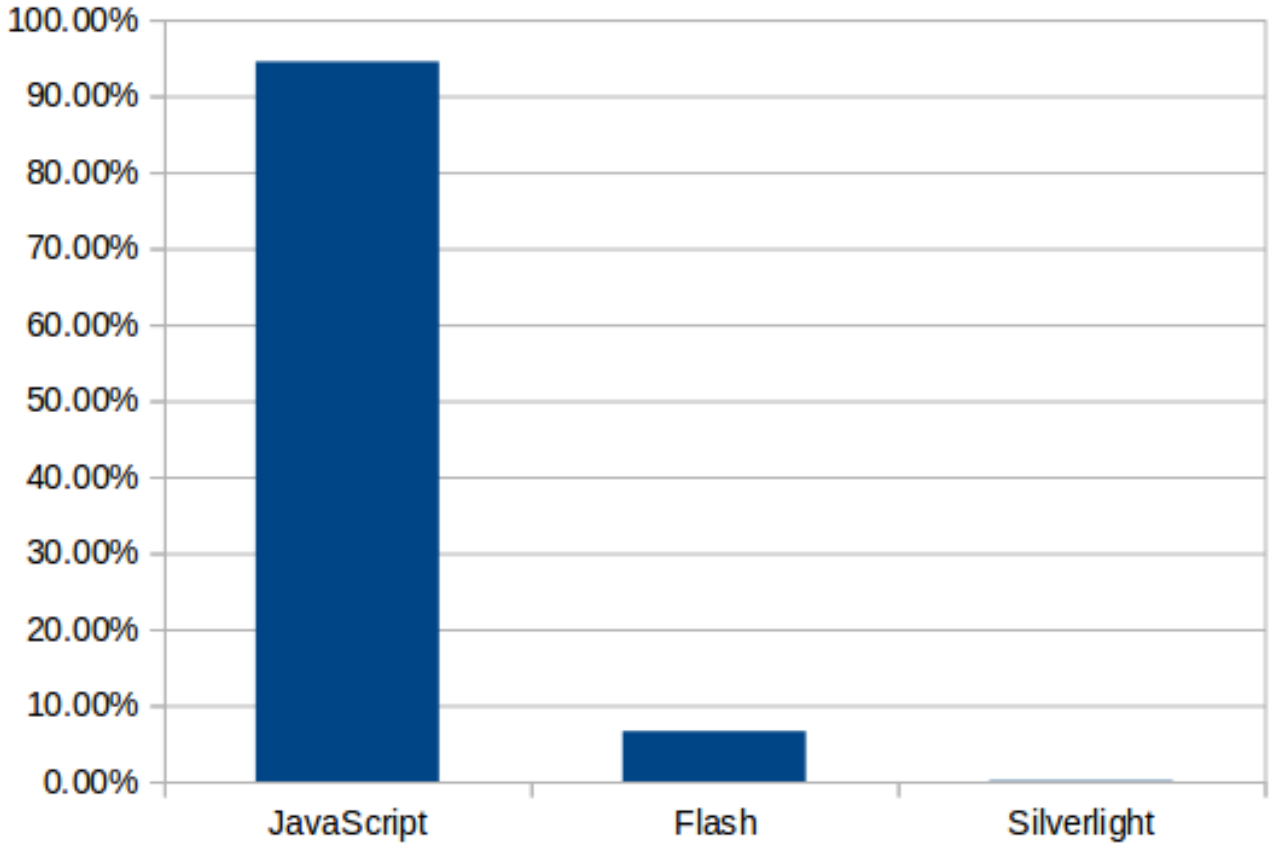
Whatever else happens...

...there will most likely be a migration path for HTML5.

2. Cascading Stylesheets

- 20 years old
- Many properties, values and terms date back to early modern printing (*font*, *italic*, *pt*, *en*, *justify*)
- Design principles incorporate *forward and backward compatibility*

3. JavaScript



https://w3techs.com/

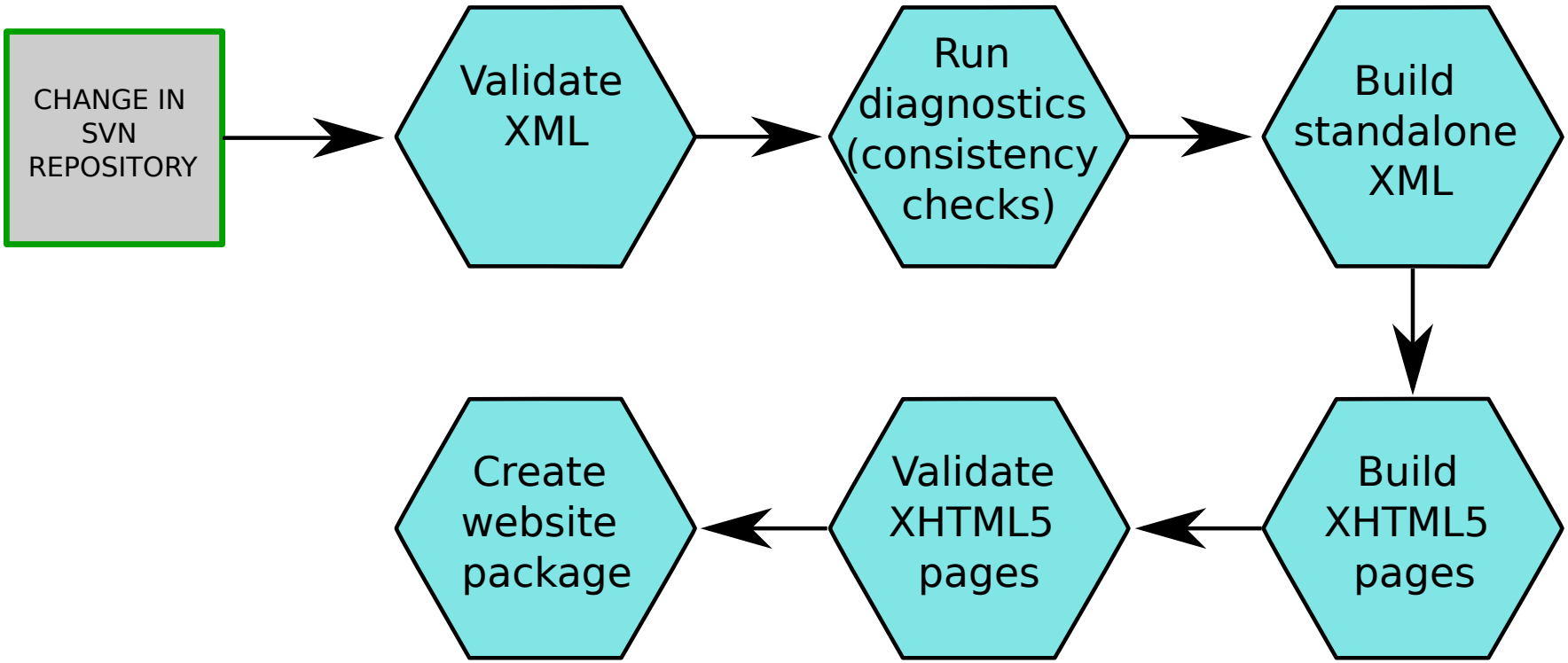
3. JavaScript

- Standardized as ECMA-262 and ISO 16262
- Committed to backwards compatibility

XHTML5 + CSS + JS

- Web pages
- Mobile apps (Android, iOS)
- EPUB3

Build process



Goals and benefits

Goals and benefits

- Site works from any web server.

Goals and benefits

- Site works from any web server.
- Site works off your hard drive.

Goals and benefits

- Site works from any web server.
- Site works off your hard drive.
- Site works off a USB drive.

Goals and benefits

- Site works from any web server.
- Site works off your hard drive.
- Site works off a USB drive.
- Site can be copied from anywhere to anywhere.

Goals and benefits

- Site works from any web server.
- Site works off your hard drive.
- Site works off a USB drive.
- Site can be copied from anywhere to anywhere.
- No JavaScript support? Site still works.

Goals and benefits

- Site works from any web server.
- Site works off your hard drive.
- Site works off a USB drive.
- Site can be copied from anywhere to anywhere.
- No JavaScript support? Site still works.
- No CSS? Site still works.

Further requirement

- Every significant entity in the site has its own page at a unique URL. Not this:

  http://graves.uvic.ca/graves/site/xbrowse.xq?collection=%2Fdb%2Fgraves&type=diaryentry&query_stored=false&action=browse&search_text=-1&day=22&month=02&year=1935

  but this:

  http://graves.uvic.ca/diary_1935-02-22.html

Implications

Implications

- You must build *every single page* your site will ever need.

Implications

- You must build *every single page* your site will ever need.
- That’s a lot of pages (7663 for the *Map of Early Modern London*).

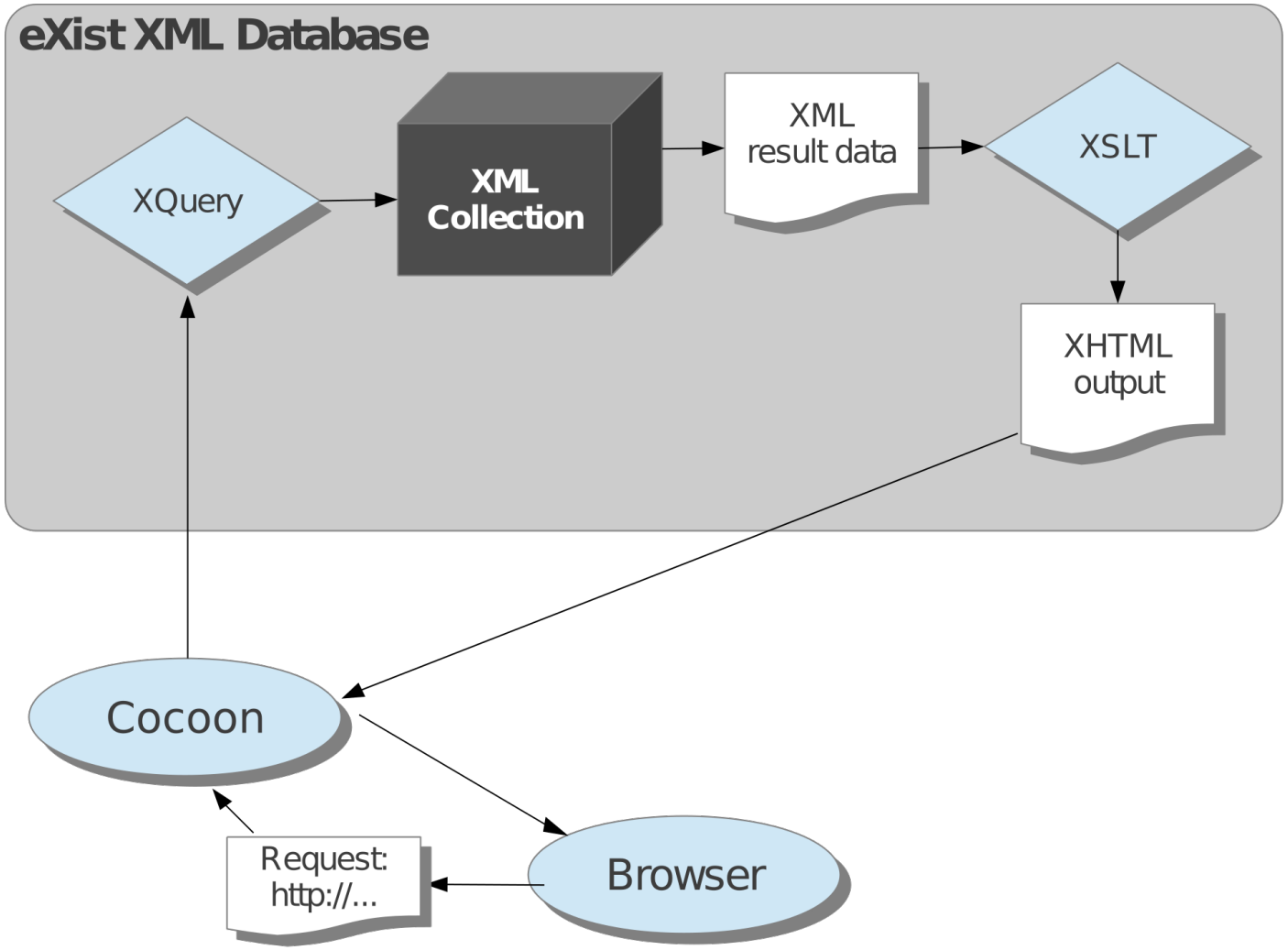
Implications

- You must build *every single page* your site will ever need.
- That's a lot of pages (7663 for the *Map of Early Modern London*).
- Every page must include *all the information* it needs to function, so lots of duplication.

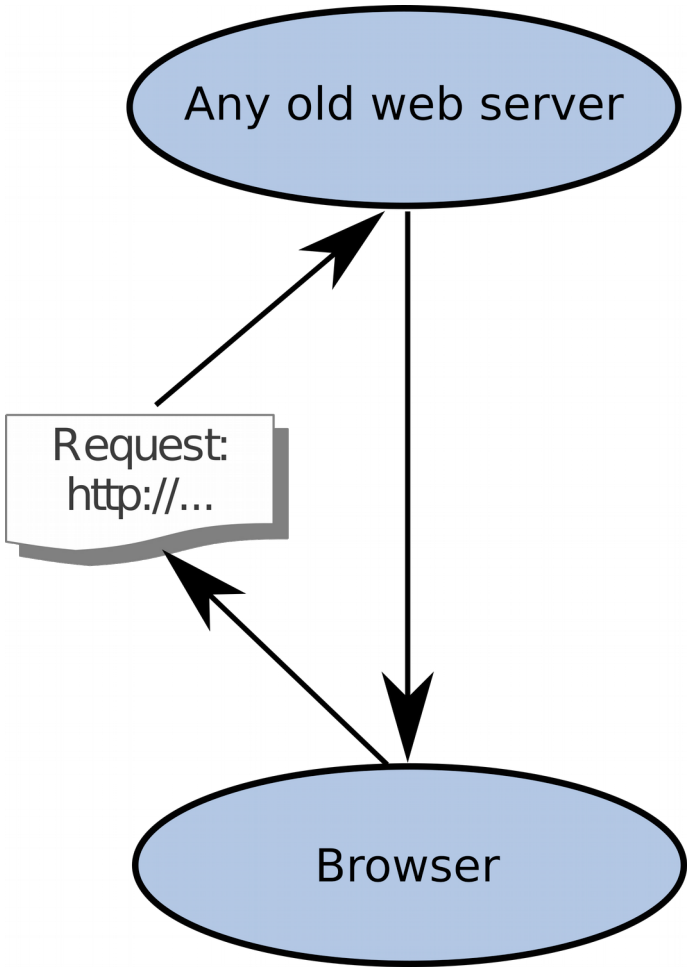
Implications

- You must build *every single page* your site will ever need.
- That's a lot of pages (7663 for the *Map of Early Modern London*).
- Every page must include *all the information* it needs to function, so lots of duplication.
- So what? Big deal. It's text.

Old model


eXist XML Database
XQuery
XML Collection
XML result data
XSLT
XHTML output
Cocoon
Request: http://...
Browser

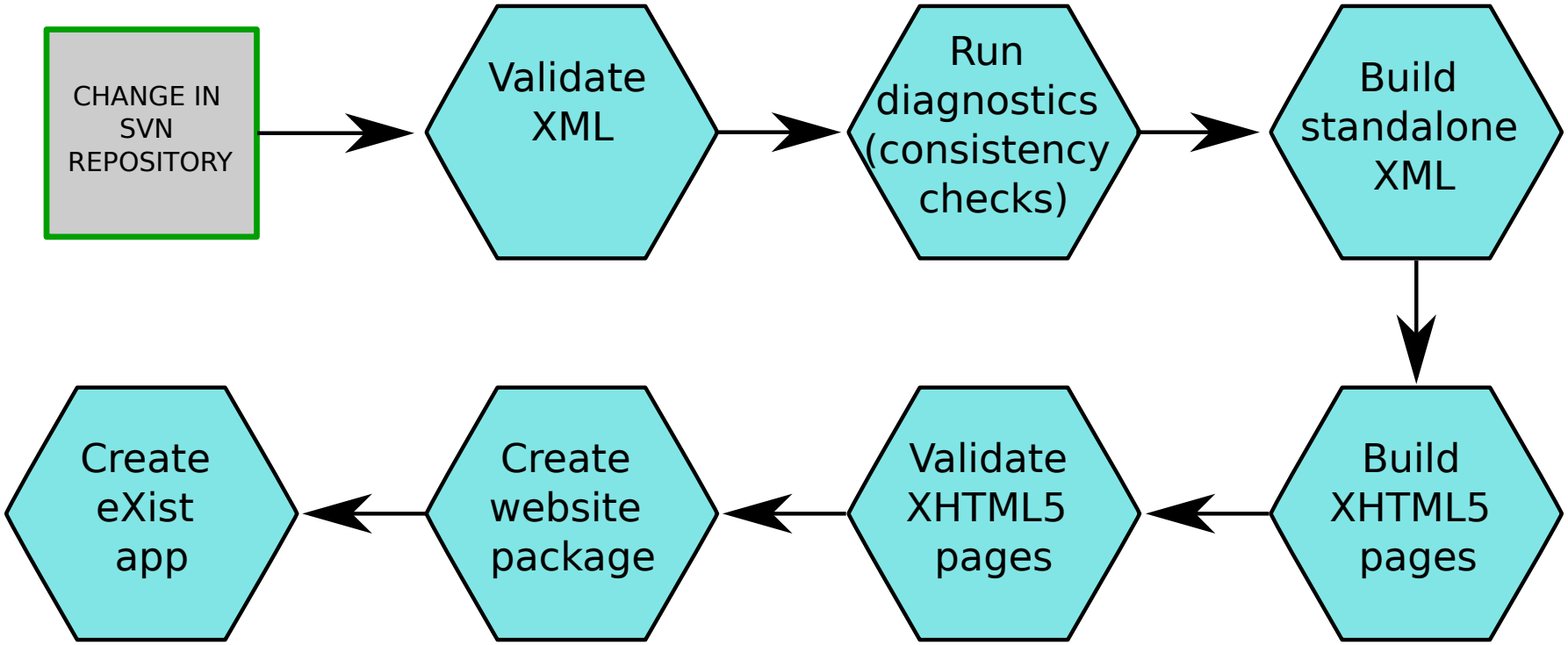
New model



What we can’t do

- Site search
- Combinatorially explosive query responses (search is a special case of this)

Back to eXist for search etc.




XHTML
CSS
JS